AMENDMENTS TO THE CLAIMS
Please amend the claims as follows:

Claims 1-9 (presently canceled)

10. (presently amended) A method for implementing a finite impulse response filter based upon ~~achieving~~ an averaged result of ~~on~~ packed binary values A_1,A_2,~~A_3,A_4,~~ the method using a $PAVG$ instruction that computes a rounded-up average on the ~~first and second sets of~~ packed values A_1 and A_2 to produce a resulting set of packed averages, wherein B ~~B_1~~ $= PAVG(A_1, A_2)$ ~~and $B2 = PAVG(A_3,A_4)$~~, the method comprising deriving a result, R, as

$$R = \begin{cases} \sim PAVG(B_1,B_2) \sim (B_1,B_2) \,\&\, ONE \text{ when } E = 0 \\ PAVG(B_1,B_2) \text{ when } E = 1 \end{cases}$$

$$(A_1 + A_2 - 2^* ONE) >> 1 = PAVG(A_1 + A_2) - ONE \cdot (A_1 \wedge A_2) \,\&\, ONE),$$

$$(A_1 + A_2 - ONE) >> 1 = CLIP(\, PAVG(A_1 + A_2) \sim ONE),$$

$$(A_1 + A_2) >> 1 = PAVG(A_1 + A_2) - (A_1 \wedge A_2) \,\&\, ONE),$$

$$(A_1 + A_2 + 2^* ONE) >> 1 = PAVG(A_1 + A_2) + (\sim (A_1 \wedge A_2) \,\&\, ONE),$$

wherein ONE is a value with a one in the least significant bit position of one or more packed values and CLIP ( ) truncates the result to the appropriate packed bits ~~wherein $E = 1$ when both (A_1+A_2+ONE) and (A_3+A_4+ONE) are odd integers~~.

Claims 11-25 (presently canceled)

26. (new) A method for implementing a finite impulse response filter based upon an averaged result of packed binary values A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, the method using a $PAVG$ instruction that computes

a rounded-up average on a first, second, third and fourth sets of packed values and subsequent averages of these rounded-up averages to produce a resulting set of packed averages, wherein

$B_1 = PAVG(A_1 + A_2)$, $B_2 = PAVG(A_3 + A_4)$, $B_3 = PAVG(A_5 + A_6)$, $B_4 = PAVG(A_7 + A_8)$, and

$C_1 = PAVG(B_1 + B_2)$, $C_2 = PAVG(B_3 + B_4)$,

the method comprising deriving a result, R, as

$R = PAVG(C_1, C_2) - ((C_1 \wedge C_2) \mid Z \mid T) \,\&\, ONE$,

wherein *ONE* is a value with a one in the least significant bit position of one or more packed values and wherein

$T = U \,\&\, V \,\&\, W \,\&\, ((EB_1 \,\&\, EB_2) \mid (EB_3 \,\&\, EB_4))$,

$EB_1 = (A_1 \wedge A_2)$, $EB_2 = (A_3 \wedge A_4)$, $EB_3 = (A_5 \wedge A_6)$, $EB_4 = (A_7 \wedge A_8)$,

$EC_1 = (B_1 \wedge B_2)$, $EC_2 = (B_3 \wedge B_4)$,

$U - EC_1 \mid EC_2$,

$V - EB_1 \mid EB_2$,

$U - EC_1 \mid EC_2$,

$W - EB_3 \mid EB_4$,

$X = V \mid W$,

$Y = U \mid X$, and

$Z = (EC_1 \,\&\, EC_2 \,\&\, X)$.

27. (new) A method for implementing a finite impulse response filter based upon an averaged result of packed binary values A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, the method using a *PAVG* instruction that computes a rounded-up average on a first, second, third and fourth sets of packed values and subsequent averages of these rounded-up averages to produce a resulting set of packed averages, wherein

$B_1 = PAVG(A_1 + A_2)$, $B_2 = PAVG(A_3 + A_4)$, $B_3 = PAVG(A_5 + A_6)$, $B_4 = PAVG(A_7 + A_8)$, and

$C_1 = PAVG(B_1 + B_2)$, $C_2 = PAVG(B_3 + B_4)$,

the method comprising deriving a result, R, as

$R = PAVG(C_1, C_2) - ((ED \mid Y) \,\&\, ONE) - (U \,\&\, V \,\&\, ED \,\&\, ONE)$,

wherein *ONE* is a value with a one in the least significant bit position of one or more packed values and wherein

$P = (EB_3 \,\&\, EB_4)$,

$U = EB_1 \,\&\, EB_2 \,\&\, P$,

$V = EC_1 \,\&\, EC_2$,

$W = (B_3 \wedge B_4)$,

$U = EB_3 \mid EB_4$,

$X = (EC_1 \mid EC_2) \,\&\, ((EB_1 \,\&\, (EB_2 \mid W)) \mid (EB_2 \,\&\, W) \mid P)$,

$Y = (X \mid V \mid U)$, and

$ED = (C_1 \wedge C_2)$.

28. (new) A method for implementing a finite impulse response filter based upon an averaged result of packed binary values A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, the method using a *PAVG* instruction that computes a rounded-up average on a first, second, third and fourth sets of packed values and subsequent averages of these rounded-up averages to produce a resulting set of packed averages, wherein

$B_1 = PAVG(A_1 + A_2)$, $B_2 = PAVG(A_3 + A_4)$, $B_3 = PAVG(A_5 + A_6)$, $B_4 = PAVG(A_7 + A_8)$, and

$C_1 = PAVG(B_1 + B_2)$, $C_2 = PAVG(B_3 + B_4)$,

the method comprising deriving a result, R, as

$R = PAVG(C_1, C_2) - (ED \mid U \mid V((EC_1 \mid EC_2) \,\&\, W)) \,\&\, ONE - ED \,\&\, U \,\&\, V \,\&\, ONE$,

wherein *ONE* is a value with a one in the least significant bit position of one or more packed values and wherein

$P = (EB_1 \mid EB_4)$,

$Q = (EB_3 \mid EB_2)$,

$U = (EB_2 \,\&\, EB_3 \,\&\, P) \mid (EB_4 \,\&\, EB_1 \,\&\, Q)$

$V = EC_1 \,\&\, EC_2$,

$W = P \mid Q$, and

$ED = (C_1 \wedge C_2)$.

29. (new) A method for implementing a finite impulse response filter based upon an averaged result of packed binary values $A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8$, the method using a $PAVG$ instruction that computes a rounded-up average on a first, second, third and fourth sets of packed values and subsequent averages of these rounded-up averages to produce a resulting set of packed averages, wherein

$B_1 = PAVG(A_1 + A_2)$, $B_2 = PAVG(A_3 + A_4)$, $B_3 = PAVG(A_5 + A_6)$, $B_4 = PAVG(A_7 + A_8)$, and

$C_1 = PAVG(B_1 + B_2)$, $C_2 = PAVG(B_3 + B_4)$,

the method comprising deriving a result, R, as

$R = PAVG(C_1, C_2) - (ED \mid U \mid W) \,\&\, ONE - ED \,\&\, ((W \,\&\, V) \mid Z) \,\&\, ONE$,

wherein ONE is a value with a one in the least significant bit position of one or more packed values and wherein

$P = (EB_3 \mid EB_4)$,

$Q = (EB_3 \mid EB_4)$,

$U = (EB_1 \,\&\, (EB_2 \mid Q)) \mid (EB_2 \,\&\, Q) \mid P$,

$V = EB_1 \,\&\, EB_2 \,\&\, P$,

$W = EC_1 \mid EC_2$,

$Z = (EC_1 \,\&\, EC_2 \,\&\, U)$, and

$ED = (C_1 \wedge C_2)$.

30. (new) A method for implementing a finite impulse response filter based upon an averaged result of packed binary values A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8,A_9, A_{10}, A_{11}, A_{12}, A_{13}, A_{14}, A_{15}, A_{16}, the method using a $PAVG$ instruction that computes a rounded-up average on a first through eighth sets of packed values and subsequent averages of these rounded-up averages to produce a resulting set of packed averages, wherein

$B_1 = PAVG(A_1 + A_2)$, $B_2 = PAVG(A_3 + A_4)$, $B_3 = PAVG(A_5 + A_6)$, $B_4 = PAVG(A_7 + A_8)$,

$B_5 = PAVG(A_9 + A_{10})$, $B_6 = PAVG(A_{11} + A_{12})$, $B_7 = PAVG(A_{13} + A_{14})$, $B_8 = PAVG(A_{15} + A_{16})$,

$C_1 = PAVG(B_1 + B_2)$, $C_2 = PAVG(B_3 + B_4)$, $C_3 = PAVG(B_5 + B_6)$, $C_4 = PAVG(B_7 + B_8)$, and

$D_1 = PAVG(C_1 + C_2)$, $D_2 = PAVG(C_3 + C_4)$,

the method comprising deriving a result, R, as

$R = PAVG(D_1, D_2) - ((ET_1 \,\&\, ET_2) \mid \sim E) \,\&\, W)) \,\&\, (ET_1 \,\&\, ET_2) \mid E)$

$\&\, ONE - (D_1 \wedge D_2) \,\&\, \sim (ET_1 \wedge ET_2 \wedge E) \,\&\, ONE$,

wherein ONE is a value with a one in the least significant bit position of one or more packed values.